aci avances en ciencias e ingenierías

Artículo/Article
Vol. 17, nro. 1
e3710

A temporal approach to urban crime forecasting using recurrent neural networks

Juan Pablo Perez Leal[1], Andrés Sebastián Ríos Gutiérrez[2 3], David Romo-Bucheli[1]

[1]Escuela Ingeniería de Sistemas e informática, Facultad de ingenierías fisicomecánicas, Universidad Industrial de Santander, Bucaramanga, Colombia.

[2]Departamento de Estadística, Universidad Nacional de Colombia, Bogotá, Colombia.

[3]Escuela de Matemáticas, Universidad Industrial de Santander

*Autor para correspondencia/Corresponding author: juanpabloperezleal@gmail.com

Un enfoque temporal para la predicción del crimen urbano usando redes neuronales recurrentes

Abstract

This study investigates the use of Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks to predict crime patterns in Bucaramanga, Colombia. A temporal approach is presented, which starts by splitting the city into 17 communes. Using a dataset of robbery incidents from 2016 to January 2023, we developed individual time series models for each commune. Then, we used the Root Mean Squared Error (RMSE) as the evaluation metric in these regression tasks. The LSTM models consistently outperformed both the RNN and ARIMA models, a classical methodology for time series prediction, achieving lower RMSE scores. The LSTM model yielded an average RMSE of 2.875 (with a standard deviation of 1.657), which is considerably lower than that obtained by the RNN model 3.101 (1.82) and the ARIMA model 3.428 (2.57). These results show that LSTM better captures the complex temporal dependencies in the data. Future work should explore hybrid models and the incorporation of additional data sources to enhance predictive accuracy further.

Keywords: Crime prediction, Neural Networks, Long Short-Term Memory (LSTM), Recurrent Neural Networks

Resumen

Este estudio investiga el uso de Redes Neuronales Recurrentes (RNN) y redes de Gran Memoria a Corto Plazo (LSTM) para predecir patrones de criminalidad en Bucaramanga, Colombia. Se presenta un enfoque temporal que comienza dividiendo la ciudad en 17 comunas. Utilizando un conjunto de datos de incidentes de robos desde 2016 hasta enero de 2023, se desarrollaron modelos de series de tiempo individuales para cada comuna. Posteriormente, se empleó el Error Cuadrático Medio (RMSE) como métrica de evaluación en estas tareas de regresión. Los modelos LSTM superaron de manera consistente tanto a los modelos RNN como a los modelos ARIMA, una metodología clásica para la predicción de series temporales, logrando menores puntajes de RMSE. El modelo LSTM obtuvo un RMSE promedio de 2.875 (con una desviación estándar de 1.657), considerablemente inferior al obtenido por el modelo RNN, con 3.101 (1.82), y el modelo ARIMA, con 3.428 (2.57). Estos resultados demuestran que LSTM captura mejor

OPEN ACCESS

Licencia Creative Commons Atribución-NoComercial 4.0

Editado por / Edited by: Dennis Cazar

Recibido / Received: 29/11/2024

Aceptado / Accepted: 09/12/2024

Publicado en línea / Published online: 12/05/2025

 DOI: https://doi.org/10.18272/aci.v17i1.3710

las complejas dependencias temporales en los datos. Trabajos futuros deberían explorar modelos híbridos y la incorporación de fuentes de datos adicionales para mejorar aún más la precisión predictiva.

Palabras clave: Predicción de crímenes, redes neuronales, redes LSTM, Redes Neuronales Recurrentes.

INTRODUCTION

Crime prediction is a critical area of study for urban planning and public safety. The ability to anticipate criminal activities enables authorities to allocate scarce resources more effectively, plan preventive measures, and enhance community safety. Traditional methods of crime prediction have often relied on statistical models and historical data analysis [1]. Some of them are related to stational ARIMA models [2, 3], and others to GARCH models [4]. However, with the advent of advanced computational techniques, there is a growing interest in leveraging machine learning algorithms to improve the accuracy and reliability of these predictions. In addition, the crime data in Bucaramanga is a large volume database as may be seen in [5], and therefore it is difficult to get a good fit by using parametric time series models [6].

In recent years, Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs), have shown significant promise in modeling time series data due to their ability to capture temporal dependencies and patterns [7, 8, 9]. RNNs, with their recurrent connections, are designed to recognize sequences and trends over time, making them suitable for analyzing temporal data. LSTMs, an extension of RNNs, address the limitations of traditional RNNs by incorporating memory cells that can store information for long periods by overcoming the issue of vanishing gradients and enabling the network to retain long-term dependencies [10].

This paper focuses on the application of RNNs and LSTMs to predict crime patterns in the city of Bucaramanga. We aim to develop models that can identify and predict the temporal trends in crime occurrences with higher accuracy. The study will explore the effectiveness of these models in capturing the intricate patterns of criminal activities, considering various temporal factors such as time of day, day of the week, and seasonal variations.

RELATED WORKS

Predicting crime in cities has become a critical area of research for law enforcement and public safety agencies. By analyzing historical crime data, researchers can identify patterns and develop tools to anticipate future occurrences. Several existing research approaches are based on various machine learning for predicting crime. For instance, in [11], the authors utilize Chicago's public crime dataset for crime trend prediction based on a LSTM model. It also incorporates POI (Point of Interest) information data and employs a convolutional model to analyze and compare the time series distribution, spatial distribution, and prediction accuracy of various algorithm models. The experimental results demonstrate a

 DOI: https://doi.org/10.18272/aci.v17i1.3710

hit rate of 20.11% for different experiments using the LSTM algorithm. Another example is applied in Atlanta, which is one of the cities with high crime rates in the United States. In [12], the authors show that crime events exhibit spatial aggregation and temporal dependence, indicating that criminal incidents are predictable. In the mentioned paper, crime data from Atlanta, spanning 2009 to 2016, is used to identify spatio-temporal distribution features and create statistical visualizations. Forecast of daily crime occurrences is obtained via a LSTM model. The study further explores the impact of varying spatio-temporal scales on prediction accuracy. When using an input time series length of 50 days and a spatial cell size of 0.05 degrees, the correlation coefficient (R value) between predicted and observed crime data exceeds 0.87. Finally, in [13] a spatio-temporal approach is applied to model monthly robberies in Buenos Aires (Argentina). The main objective of the study is to implement predictive models to associate specific corners and times with the occurrence of crimes, based on monthly robbery records in Buenos Aires from 2017 to 2020. Initially, time series models using only historical crime counts are employed to predict criminal incidents at each street corner. Additionally, predictions include covariates related to the surrounding environment and weather conditions, transforming the problem into a spatio-temporal study. The study shows that crime, being a human activity, does not follow random behavior but rather social patterns influenced by time, weather, and geographical zones; in some cases, it shows an impact in the model of more than 13% for some features [13].

MATERIALS

Bucaramanga is the capital of the department of Santander, Colombia. It is located on a plateau at 960 meters above sea level, with a population of approximately 600,000. Nestled in the eastern Andes Mountains of Colombia, Bucaramanga is characterized by its undulating topography and dense urbanization. The city extends over a series of hills and valleys, resulting in variations in accessibility and visibility across different areas [14]. Administratively, Bucaramanga is divided into 17 communes, each further subdivided into neighborhoods. Leveraging the big sizes and geographical locations of these communes, we get a framework to identify patterns and trends in crime rates. The dataset used in this study encompasses robbery incidents recorded from 2016 to January 2023. This data was collected by Bucaramanga's city hall, in collaboration with the municipal police, and is publicly available on the Datos Abiertos platform [5]. The collected data was grouped in weeks of the year instead of daily, resulting in the count of robberies for each commune. The period associated with the training data spanned from January 1, 2016, to September 12, 2022. A total of 17 time series were obtained, one for each of the communes. Test data for each commune was also extracted and is associated with the period spanning from September 13, 2022, to November 20, 2022

Each record in the dataset includes the date of the robbery and the specific commune in which it occurred.

This dataset allows for the creation of a time series capturing the frequency of robbery incidents within each commune of Bucaramanga. Given the nature of the problem, it is expected that there will be days with zero reported incidents in some communes. This variability provides an opportunity to explore temporal patterns and anomalies in criminal activity across different areas of the city.

 DOI: https://doi.org/10.18272/aci.v17i1.3710

The collected data was grouped in weeks of the year instead of daily. The period associated with the training data spanned from January 1, 2016, to September 12, 2022. The models were trained for each commune independently, allowing them to learn the underlying patterns and trends specific to each commune in Bucaramanga. The test data is associated with the period spanning from September 13, 2022, to November 20, 2022. In Figure 1, a heatmap of Bucaramanga displaying the total count of robberies during the training period is shown. The crime distribution shows remarkable differences across the different communes in Bucaramanga.

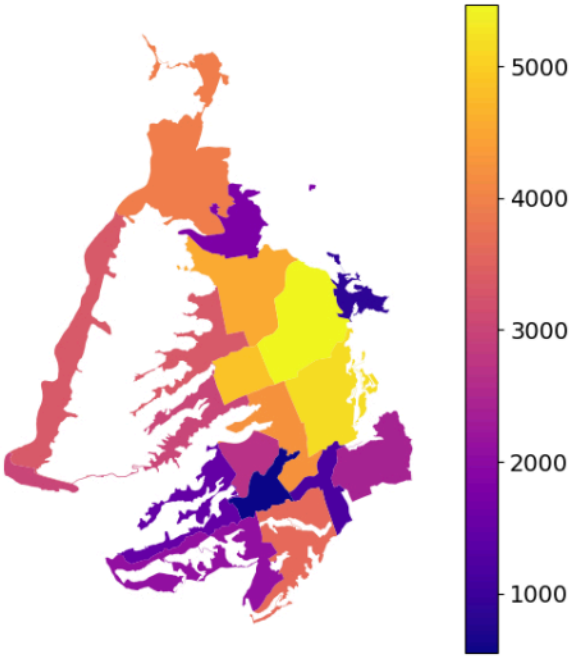

FIGURE 1. Map of Bucaramanga displaying the spatial distribution and count of robbery incidents from January 2016 to November 2022. Own elaboration

METHODOLOGY

In this work, we compared several methodologies for crime prediction, utilizing an independent temporal approach for each of the communes of Bucaramanga. Our purpose is to evaluate the effectiveness of different time series models in forecasting crime patterns. These are the different models that are going to be employed in this study.

ARIMA: Among the most commonly used tools for time series forecasting is the ARIMA model (Autoregressive Integrated Moving Average). The ARIMA model is valuable in situations where the data exhibits autocorrelation, that is, correlation between

DOI: https://doi.org/10.18272/aci.v17i1.3710

past random variables and the present time random variable. Additionally, it can handle time series that are non-stationary, i.e., those that show trends or systematic variations over time [15]. It is important to note that the ARIMA model is a dynamic time series model, meaning that future estimates are explained by past data and not by independent variables [16, 17]. The ARIMA model can be defined as a composition of three main components: Autoregression (AR), Integration (I), and Moving Average (MA). Autoregression (AR) refers to the linear dependency of a current observation on past observations. In an AR model, the dependent variable is linearly regressed on its past values up to order p.

$$x_t = c + \phi_1 x_{t-1} + \phi_2 x_{t-2} + \cdots + \phi_p x_{t-p} + \epsilon_t, \tag{1}$$

where x_t is the dependent variable at time t, c is a constant, $\phi_1, \phi_2, \ldots, \phi_p$ are the autoregression parameters, and ϵ_t is a random error term at time t. Generally, this follows a white noise (WN) process with variance σ^2, for all $t \geq 0$, $\epsilon_t\ WN\,(0, \sigma^2)$ [17].

The Moving Average (MA) models the relationship between an observation and a residual error term generated by a moving average of errors in previous periods. In an MA model, the dependent variable is linearly regressed on past error terms up to order q. The relationship is mathematically defined as:

$$x_t = c + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q} + \epsilon_t,\ \epsilon_t \sim WN\,(0, \sigma^2), \tag{2}$$

where c is a constant, $\theta_1, \theta_2, \ldots, \theta_q$ are the moving average parameters, and $\epsilon_1, \epsilon_2, \ldots, \epsilon_{t-q}$ are the error terms at previous times. Integration (I) refers to the process of making a time series stationary, i.e., removing trends or systematic patterns that may affect the prediction. This is achieved by differencing the time series until it becomes stationary. The order of differencing is denoted as d. This difference is recurrently defined as:

$$\Delta^d x_t = \Delta^{d-1} x_t - \Delta^{d-1} x_{t-1},\ \Delta x_t = \Delta^1 x_t = x_t - x_{t-1} \text{ and } d = 2, 3, \ldots \tag{3}$$

The ARIMA model combines these three components into a single framework and is denoted as ARIMA(p, d, q), where p is the order of autoregression, d is the order of differencing to get a stationary process (see [18]), and q is the order of moving average.

Delay operator is defined, for all u an integer number, as $L^u x_t := x_{t-u}$ [18]. Based on this operator, the ARIMA(p, d, q) model is defined by:

$$\Phi_p(L)(1-L)^d x_t = c + \Theta_q(L)\epsilon_t \text{ with } \epsilon_t \sim WN\,(0, \sigma^2), \tag{4}$$

where $\Phi_p(L)$, $\Theta_q(L)$ are polynomials, respectively called the autoregressive polynomial and the moving average polynomial. In consequence, d represents the order of the differences required for the stochastic process $\Phi_p(L)x_t = c + \Theta_q(L)\epsilon_t$ to be stationary.

RNN Model: Recurrent neural networks (RNNs) are designed to recognize patterns in sequences of data, making them well-suited for tasks involving time series analysis. Unlike traditional feedforward neural networks, RNNs have connections that form directed cycles, enabling them to maintain a memory of previous inputs [19].

 DOI: https://doi.org/10.18272/aci.v17i1.3710

The core of an RNN is the recurrent cell, which processes the input sequence one step at a time while maintaining a hidden state that evolves over time. We can do a mathematical formulation of this for an input sequence $x = (x_1, x_2, ..., x_t)$ where x_t represents the input at time step t. In our case, x_t would be the number of incidents at date t [20].

There is a hidden state h_t at time step t that is computed using the current input x_t and the previous hidden state h_{t-1}:

$$h_t = \sigma(W_{hx} x_t + W_{hh} h_{t-1} + b_h) \tag{5}$$

where W_{hx} is the weight matrix connecting the input to the hidden state, W_{hh} is the weight matrix connecting the hidden state to itself from the previous time step, b_h is the bias vector, and σ is a nonlinear activation function, typically a hyperbolic tangent or a ReLU function [21]. The output $\hat{y}_t$ at time step t is computed by using the current hidden state h_t:

$$t = g(W_{hy} h_t + b_y) \tag{6}$$

where W_{hy} is the weight matrix connecting the hidden state to the output, b_y is the bias vector, and g is typically a linear or softmax function depending on the task. In our case, g is a linear function as we are dealing with regression. The RNN model learns to adjust the weights W_{hx}, W_{hh}, and W_{hy}, as well as the biases b_h and b_y, by minimizing a loss function over the training data. For time series prediction, a common choice is the mean squared error (MSE).

In the context of this study, we employ RNNs to predict crime patterns based on a time series dataset. Each time series contains two columns: one for the number of robbery incidents and the other for the corresponding date. The same process is applied separately for each of the 17 communes in Bucaramanga, resulting in 17 distinct models, each tailored to the specific crime dynamics of its respective commune.

LSTM Model: LSTM networks are a type of RNN that address the limitations of traditional RNNs, particularly the issue of vanishing and exploding gradients. LSTMs introduce a more complex unit structure compared to traditional RNNs, featuring memory cells that can store information for extended periods. The key components of an LSTM cell include the cell state (c_t) and three types of gates: input gate, forget gate, and output gate. These gates regulate the flow of information into, out of, and within the cell [22].

In this context, given the number of incidents on date t for an input sequence $x = (x_1, x_2, ..., x_t)$, the LSTM network processes this information in a structured manner.

The forget gate determines what portion of the previous cell state (c_{t-1}) should be retained based on the current input (x_t) and the previous hidden state (h_{t-1}):

$$f_t = \sigma\,(W_f x_t + U_f h_{t-1} + b_f) \tag{7}$$

where σ is the sigmoid function, W_f and U_f are weight matrices, h_{t-1} is the previous hidden state, and b_f is the bias vector.

 DOI: https://doi.org/10.18272/aci.v17i1.3710

The input gate controls the amount of new information added to the cell state. It produces an intermediate candidate cell state (h_t) that is regulated by the input gate (i_t):

$$i_t = \sigma\,(W_i x_t + U_i h_{t-1} + b_i) \qquad \tilde{c}_t = \tanh\,(W_c x_t + U_c h_{t-1} + b_c) \tag{8}$$

where i_t is the input gate output, W_i is the weight matrix for the input x_t, U_i is the weight matrix for the previous hidden state h_{t-1}, b_i is the bias vector for the input gate, $_t$ is the candidate cell state, W_c is the weight matrix for the input x_t related to the candidate cell state, U_c is the weight matrix for the previous hidden state h_{t-1}, and b_c is the bias vector for the candidate cell state. The cell state is updated by combining the previous cell state and the candidate cell state:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{9}$$

where $\odot$ denotes the element-wise multiplication.

The output gate determines the output of the LSTM cell based on the up-dated cell state (c_t) and the current input, producing the hidden state (h_t):

$$o_t = \boldsymbol{\sigma}\,(W_o x_t + U_o h_{t-1} + b_o) \tag{10}$$

$$h_t = o_t \odot \tanh\,(c_t)$$

where o_t is the output gate output and h_t is the hidden state at time step t. In this study, we employ LSTMs to predict crime patterns using time series data, where each time series consists of two columns: the number of robbery incidents and the corresponding date. This process is replicated for each of the 17 communes in Bucaramanga, resulting in 17 individual models tailored to the unique crime dynamics of each commune.

Each LSTM model is trained on the historical time series data of robbery incidents for a specific commune. The training process involves backpropagation through time (BPTT) to update the model parameters. Given the sparse nature of the data, with many days having zero incidents, the models must learn to identify underlying patterns and trends that contribute to the temporal dynamics of crime within each commune.

The tests were conducted using TensorFlow Keras in a Google Colab environment, utilizing Google cloud TPU v2 hardware. This setup provided high-performance computing for training deep learning models efficiently on large datasets.

Metrics: The performance of the planned models in predicting crime patterns is evaluated using the Root Mean Squared Error (RMSE). RMSE is a commonly used metric for regression tasks, and it is defined as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{11}$$

 DOI: https://doi.org/10.18272/aci.v17i1.3710

where n is the number of observations, $(\hat{y})$ is the predicted value for the i-th observation, and y_i is the actual value for the i-th observation. Lower RMSE values indicate a better fit of the model to the data, as it implies that the differences between the predicted and actual values are smaller.

We also computed the SMAPE metric defined as:

$$SMAPE = \frac{100\%}{n} \sum_{t=1}^{n} \frac{|y_t - \hat{y}_t|}{\frac{|y_t| + |\hat{y}_t|}{2}} \tag{12}$$

where n corresponds to the total number of observations, t is the index of the observation, y_t is the actual value at time t, and $\hat{y}$ is the predicted value at time t.

The SMAPE metric outputs a percentage value that quantifies the accuracy of a prediction model. The value ranges from 0% to 200%, where 0% indicates a perfect prediction with no error, meaning the predicted values match the actual values exactly. As the SMAPE value increases, it indicates a larger discrepancy between the predicted and actual values, with 200% representing the maximum possible error when one value is positive and the other is zero. A SMAPE close to 100% suggests that the model's predictions are, on average, as far from the actual values as they are close, implying poor predictive performance.

EXPERIMENTAL SETUP AND RESULTS

The RNN implementation in this study is based on the foundational work by [23]. This basic RNN model serves as a reference point, as it does not include the advanced gating mechanisms introduced in more modern architectures like LSTM and GRU. While both LSTM and GRU enhance the basic RNN architecture by incorporating gates that effectively manage memory and filter information, we chose to focus exclusively on LSTM as it is representative of this kind of RNN.

After training the models using data from each commune, the results of the previously mentioned models were compared. Figure 2 (left) shows a table of RMSE results for each commune.

The LSTM model demonstrates superior overall performance across all models evaluated in this study. It consistently achieves lower RMSE scores compared to the RNN and ARIMA models, indicating its effectiveness in capturing the complex temporal patterns in the crime data. However, it is noteworthy that the RNN model occasionally outperforms the LSTM in certain specific communes. In contrast, the ARIMA model yields higher RMSE scores, underscoring the limitations of a purely statistical, regressive approach for this type of problem.

 DOI: https://doi.org/10.18272/aci.v17i1.3710

Table 1. RMSE for each one of the models used. Best values are highlighted in bold.

Commune	ARIMA	RNN	LSTM
1	2.915	2.722	2.761
2	1.486	2.387	1.534
3	7.162	5.684	5.216
4	2.338	2.266	2.292
5	2.269	2.416	2.158
6	4.184	3.733	3.852
7	1.841	1.620	1.591
8	0.854	1.109	1.108
9	1.262	1.517	1.578
10	4.656	4.354	4.016
11	2.205	2.106	2.358
12	8.643	5.693	4.893
13	8.177	6.697	5.867
14	1.169	1.121	1.006
15	5.917	5.485	5.501
16	1.501	1.302	1.500
17	1.703	2.501	1.641
Mean	**3.428**	**3.101**	**2.875**
St. dev.	**2.570**	**1.820**	**1.657**

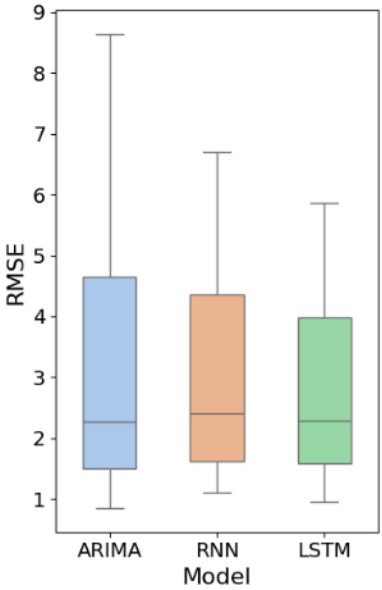

FIGURE 2. Boxplot comparison of RMSE for all the validated models. Own creation

 DOI: https://doi.org/10.18272/aci.v17i1.3710

Furthermore, the standard deviation of the RMSE scores reveals that the LSTM model not only provides more accurate predictions on average but also exhibits lower variability. This suggests a more stable and reliable predictive capability. Figure 2 and Table 1 offers a visual representation of the RMSE score distributions across the different models.

In addition to numerical evaluation, we can compare the performance of the models through visualizations of the time series predictions. Figures 3, 4, and 5 display the predicted values for ARIMA, RNN, and LSTM models alongside the actual data.

The graphical representation of the ARIMA model's predictions reveals a significant limitation: its inability to accurately predict spikes in the data. The ARIMA model tends to generate a single smooth approximation throughout the entire time series, failing to capture the abrupt changes and peaks that characterize real-world crime data. This indicates that the ARIMA model, being a purely statistical approach, lacks the flexibility needed to adapt to the complex and nonlinear patterns inherent in the crime time series.

In contrast, the RNN and LSTM models exhibit a markedly improved capacity for approximating the real data. Both models demonstrate a better alignment with the observed trends and are more adept at predicting some of the spikes in the crime incidents. The RNN and LSTM models, leveraging their recurrent structures, can retain and utilize information from previous time steps, allowing them to capture both short-term fluctuations and long-term trends more effectively.

The RNN model tends to produce smoother predictions, often missing the sharp fluctuations in the real data. This smoothing effect results in a more generalized prediction that fails to capture the variability present in the actual crime data. The RNN model struggles to detect sudden spikes in robbery counts, leading to underestimations during periods of high activity, particularly noticeable in Communes 2, 5, and 12. While the RNN model aligns with the general direction of the trends in some communes, it consistently underperforms in accurately following the actual data patterns, particularly in areas with more volatile crime rates. On the other hand, the LSTM model provides a much closer fit to the actual data compared to the RNN. It captures the fluctuations and trends more accurately, indicating its ability to model the temporal dependencies more effectively. Unlike the RNN, the LSTM model is better at detecting and replicating spikes in crime data, as seen in Communes 3, 7, and 13. This suggests that the LSTM model is more responsive to changes in the crime patterns. The LSTM model demonstrates a superior ability to track the real data across all communes. It reduces the prediction error and better follows the actual robbery counts, making it a more reliable model for this type of prediction.

Based on the analysis of SMAPE scores the models across the communes, both models exhibit similar overall performance, with average SMAPE scores of 48.02 for ARIMA and 48.59 for LSTM. However, closer inspection reveals that LSTM outperforms ARIMA in certain contexts. Specifically, in communes 1, 3, 7 and 12, the LSTM model shows superior predictive accuracy. These areas tend to have more stable or predictable crime patterns, or they exhibit higher volumes of crime, allowing LSTM's ability to capture long-term dependencies and handle complex temporal trends to shine. On the other hand, in communes where crime patterns have low variation or are closer to the mean, ARIMA tends to perform better, likely due to its strength in modeling simpler, more linear time series data.

DOI: https://doi.org/10.18272/aci.v17i1.3710

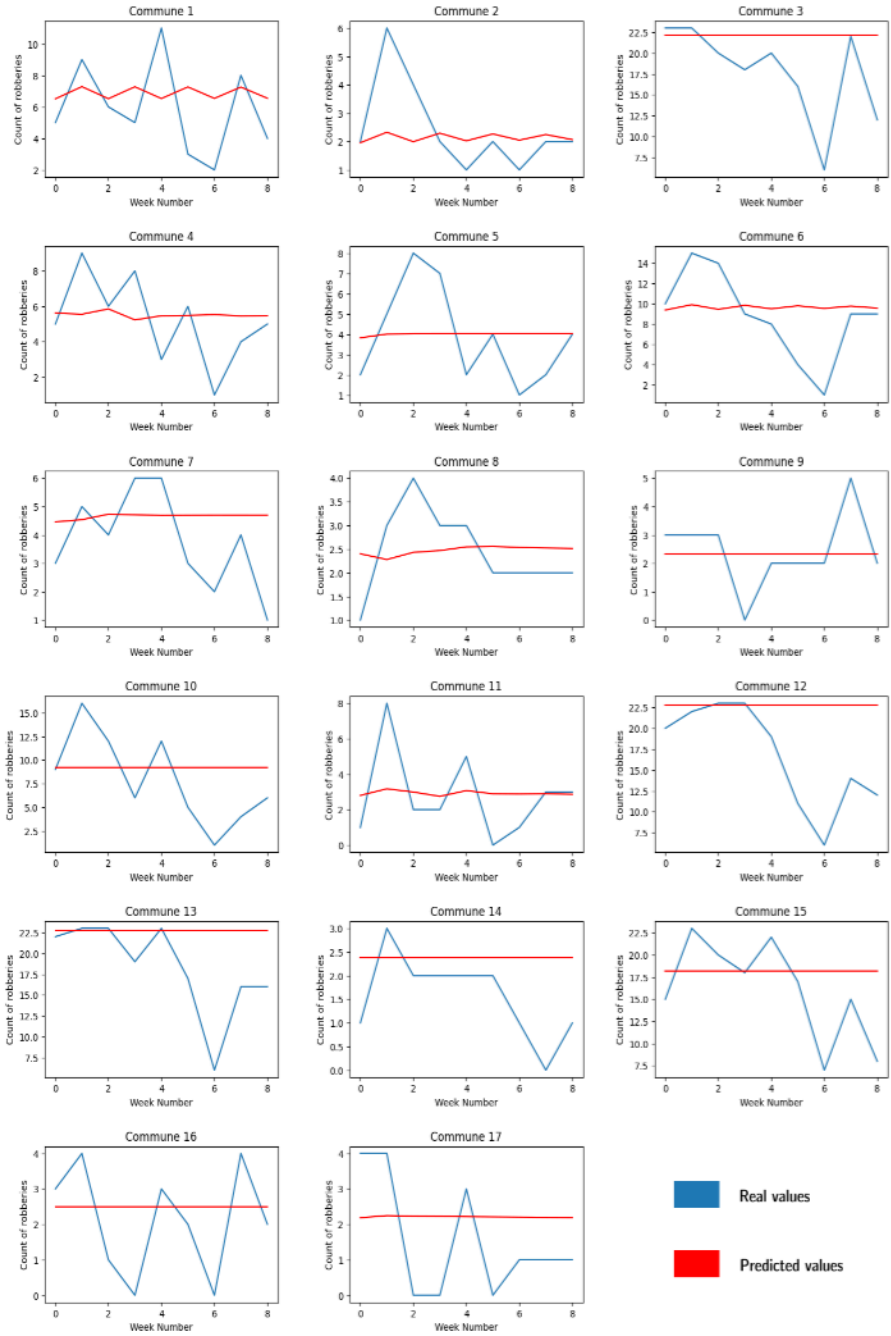

FIGURE 3. Time series of the predicted values for the ARIMA model. Own creation

DOI: https://doi.org/10.18272/aci.v17i1.3710

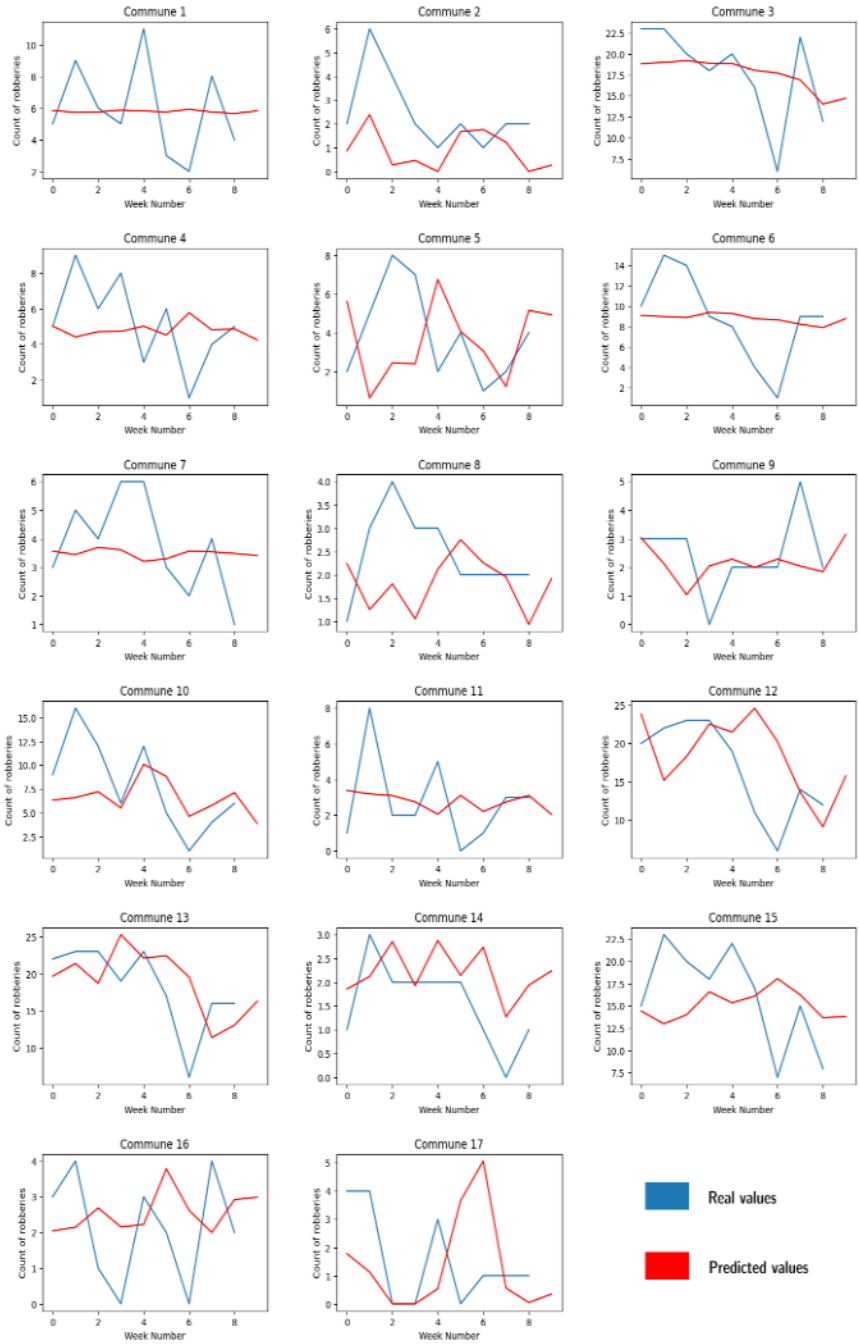

FIGURE 4. Time series of the predicted values for the RNN model. Own creation

DOI: https://doi.org/10.18272/aci.v17i1.3710

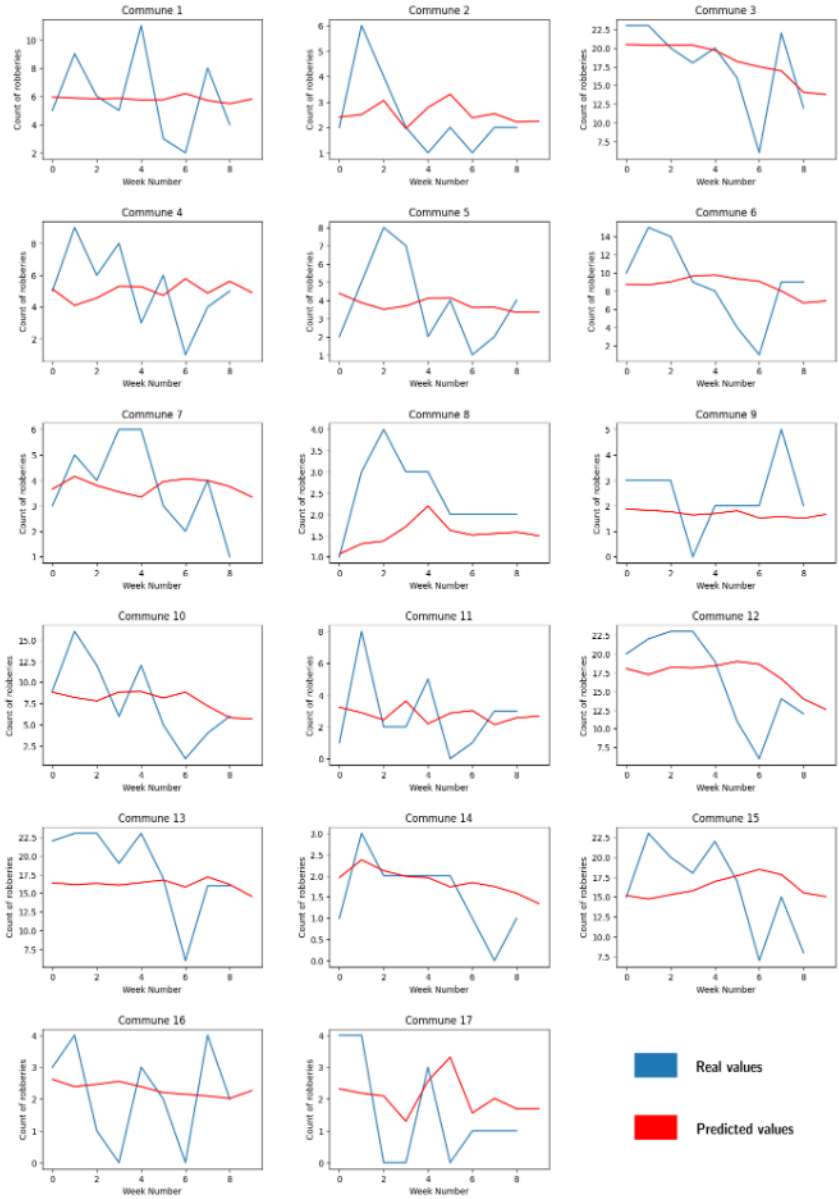

FIGURE 5. Time series of the predicted values for the LSTM model. Own creation

CONCLUSION

In this study, we explored the potential of utilizing solely temporal data to understand and predict crime patterns in a Latin American urban context, specifically Bucaramanga.

 DOI: https://doi.org/10.18272/aci.v17i1.3710

This approach underscores a significant advancement in the field of urban crime analysis. By focusing on temporal dynamics, we demonstrated that effective predictions could still be achieved without the inclusion of spatial data, which is often considered crucial in such analyses. This finding is particularly valuable for regions where spatial data may be incomplete or unavailable. Our results highlight the robustness of temporal models, such as LSTM, in extracting meaningful patterns from time-series data alone, offering a promising avenue for enhancing public security strategies in similar urban settings. Integrating High-Performance Computing (HPC) enables us to leverage real-time data for scalable AI predictions, enhancing our capacity to handle extensive datasets efficiently and effectively in dynamic environments.

The results demonstrated that LSTM models consistently outperform both RNN and ARIMA models in predicting crime incidents across all 17 communes in our case study.

Visual comparison through time series plots further elucidated the strengths and weaknesses of each model. The ARIMA model's predictions were notably smooth and failed to capture the abrupt changes in crime incidents, indicating a significant shortfall in its predictive capability. In contrast, both the RNN and LSTM models exhibited a better fit to the actual data, more accurately reflecting the observed trends and spikes.

While the study demonstrates the effectiveness of LSTMs in crime prediction, there are limitations that warrant further investigation. Incorporating more features, such as socioeconomic indicators, weather conditions, and public events, could potentially improve the model's predictive power. Future work should explore the integration of spatial data, other data sources, and the development of hybrid models to capitalize on the strengths of different neural network architectures.

AUTHORS' CONTRIBUTIONS

Writing: Juan Pablo Perez Leal, Andrés Sebastián Ríos Gutiérrez, David Romo-Bucheli
Visualization: Juan Pablo Perez Leal
Conceptualization and Supervision: Andrés Sebastián Ríos Gutiérrez, David Romo-Bucheli

CONFLICT OF INTEREST

The authors have no competing interests to declare that are relevant to the content of this article.

ACKNOWLEDGEMENT

The authors would like to thank the Vicerrectoría de Investigación y Extensión (VIE) of the Universidad Industrial de Santander for supporting this research work through the project "Adaptación de Dominio para Modelos de Aprendizaje Automático en Patología Digital con Imágenes Histológicas de Baja Magnificación," with code SIVIE 3951.

 DOI: https://doi.org/10.18272/aci.v17i1.3710

REFERENCES

[1] Thomas, A., & Sobhana, N. (2022). A survey on crime analysis and prediction. *Materials Today: Proceedings*, 58, 310–315. https://doi.org/10.1016/j.matpr.2022.02.170

[2] Ghani, U., Toth, P., & David, F. (2023). Predictive choropleth maps using ARIMA time series forecasting for crime rates in Visegrád group countries. *Sustainability, 15*(10), 8088. https://doi.org/10.3390/su15108088

[3] Noor, T. H., Almars, A. M., Alwateer, M., Almaliki, M., Gad, I., & Atlam, E. S. (2022). SARIMA: A seasonal autoregressive integrated moving average model for crime analysis in Saudi Arabia. *Electronics, 11*(23), 3986. https://doi.org/10.3390/electronics11233986

[4] Escudero, I., Angulo, J. M., & Mateu, J. (2022). A spatially correlated model with generalized autoregressive conditionally heteroskedastic structure for counts of crimes. *Entropy, 24*(7), 892. https://doi.org/10.3390/e24070892

[5] *Delitos ocurridos en el Municipio de Bucaramanga | Datos Abiertos Colombia.* (2018, October 22). Retrieved October 20, 2023 from https://www.datos.gov.co/Seguridad-y-Defensa/Delitos-ocurridos-en-el-Municipio-de-Bucaramanga/75fz-q98y/about_data

[6] Fan, J., & Yao, Q. (2008). *Nonlinear time series: Nonparametric and parametric methods.* Springer Science & Business Media.

[7] Al-Selwi, S. M., Hassan, M. F., Abdulkadir, S. J., Muneer, A., Sumiea, E. H., Alqushaibi, A., & Ragab, M. G. (2024). RNN-LSTM: From applications to modeling techniques and beyond—Systematic review. *Journal of King Saud University-Computer and Information Sciences, 36*(5), 102068. https://doi.org/10.1016/j.jksuci.2024.102068

[8] Wang, Q., Guo, Y., Yu, L., & Li, P. (2017) Earthquake prediction based on spatio-temporal data mining: An LSTM network approach. *IEEE Transactions on Emerging Topics in Computing, 8*(1), 148–158. https://doi.org/10.1109/TETC.2017.2699169

[9] Zhou, S. K., Rueckert, D., & Fichtinger, G. (Eds.). (2019). *Handbook of medical image computing and computer assisted intervention.* Academic Press.

[10] Zhang, J., Zeng, Y., & Starly, B. (2021). Recurrent neural networks with long term temporal dependencies in machine tool wear diagnosis and prognosis. *SN Applied Sciences, 3*(4), 442. https://doi.org/10.1007/s42452-021-04427-5

[11] Jiang, N., Miao, K., Chai, Y., Lu, D., & Wu, J. (2023). Spatio-temporal prediction of crime based on data mining and LSTM network. *2023 IEEE 6th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 6*, 672–676. https://doi.org/10.1109/ITNEC56291.2023.10081985

[12] Wang, S., & Yuan, K. (2019). Spatiotemporal analysis and prediction of crime events in Atlanta using deep learning. *2019 IEEE 4th International Conference on Image, Vision and Computing (ICIVC)*, 346–350. http://dx.doi.org/10.1109/ICIVC47709.2019.8981090

[13] Zambrano, R. (2022). Un enfoque Espacio Temporal para la predicción de delitos en la ciudad de Buenos Aires. *Revista de Investigación de Modelos Matemáticos Aplicados a la Gestión y a la Economía, 2*, 38–62. http://www.economicas.uba.ar/institutos_y_centros/revista-modelos-matematicos/

[14] *Alcaldía de Bucaramanga: Plan Integral de Seguridad y Convivencia para una Bucaramanga Segura 2020-2023*. (2020). https://www.bomberosdebucaramanga.gov.co/contenido/wp-content/uploads/2023/04/PISCC-Bucaramanga-2020-2023.pdf

[15] Uma Devi, B., Sundar, D., & Alli, P. (2013). An effective time series analysis for stock trend prediction using ARIMA model for nifty midcap-50. *International Journal of Data Mining & Knowledge Management Process, 3*(1), 65-78. http://dx.doi.org/10.5121/ijdkp.2013.3106

[16] Brockwell, P. J., & Davis, R. A. (1991). *Time Series: Theory and Methods.* Springer Science & Business Media.

[17] Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and Practice.* OTexts.

[18] Resnick, S. I. (2013). *Adventures in Stochastic Processes*. Springer Science & Business Media.

[19] Peña, D. (2005). *Análisis de Series Temporales*. Alianza Editorial.

[20] Lipton, Z. C., Berkowitz, J., & Elkan, C. (2015). A Critical Review of Recurrent Neural Networks for Sequence Learning. https://arxiv.org/abs/1506.00019

 DOI: https://doi.org/10.18272/aci.v17i1.3710

[21] Li, J., Xu, H., Deng, J., & Sun, X. (2016). Hyperbolic linear units for deep convolutional neural networks. *2016 International Joint Conference on Neural Networks (IJCNN)*, 353-359. https://doi.org/10.1109/IJCNN.2016.7727220

[22] Gers, F., Schmidhuber, J., & Cummins, F. (1999). Learning to forget: Continual prediction with LSTM. *1999 Ninth International Conference on Artificial Neural Networks ICANN99*, *2*, 850–855. https://doi.org/10.1049/cp:19991218

[23] Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211. https://doi.org/10.1207/s15516709cog1402_1